FORMAL TOOL USAGE AT BROADCOM CAMBRIDGE

THURSDAY, MAY 23, 2013

Chris Brown

AGENDA

- **How do we use formal verification tools in Cambridge?**

- **Why don’t we use formal tools more often?**

- **Questions?**

BROADCOM IN CAMBRIDGE

- Is an IP and chip-design center.
- There are approximately 150 employees on site.
- We are part of Broadcom’s Mobile Product Solutions BU.
- We create market-leading video, imaging, and 3D solutions for use in mobile application processors.
- We are the home of the “Raspberry Pi.”

FORMAL TOOL USAGE

- **Our most successful use of formal tools to date was in diagnosing an issue during initial silicon bring-up.**
  - A very rare failure occurred.
  - It couldn’t possibly happen, but it did.
  - We input the failure data to a formal tool, and it was able to point us in the direction of the issue.
  - Presented by David Lewsey at Club Formal UK December 2011.

CACHE BUG FOUND WITH FORMAL TOOL

- **Symptoms**
  - Writes go missing (rather than going to the wrong place).
  - Repeated polling of a shared structure is common to most occurrences.
  - Both processors writing to the same line seems to be necessary.
- **The assertion:**

```
property only_one_snoop_hit;
        @(posedge clk) ~(snoop_hit0 & snoop_hit1);
endproperty


only_one_snoop_hit_ok: assert property (only_one_snoop_hit);
```

- **Result**
  - Couple of hours runtime (including all the other assertions being checked).
  - Debug waveform is 15 clock cycles long.

CACHE BUG FOUND WITH FORMAL

- **Bug**
  - The stall-detection logic just shown checked only for tag lookups to the same address in consecutive cycles.
  - It did not account for stalls further down the pipeline.
- **It really does work.**
  - Formal tools found a problem that we had been unable to capture in simulation.
  - Debug is easy — the formal tool gave the shortest possible trace.
  - The time to set up the tool was comparable to the time to create a new block-level test bench.

CURRENT USAGE OF FORMAL TOOLS

- **Day-to-day usage**
  - A fairly small select group of design engineers successfully use formal tools to analyze their designs.
  - Usually where a standard bus protocol is in use.

- **If we have had great success with formal tools, why don’t we use them all the time?**

WHY DON’T WE MAKE MORE USE OF FORMAL TOOLS?

- **Three classes of reasons:**
  - Excuses
  - Real (?) technical
  - Cost

EXCUSES

- **Excuses are usually based on hearsay and not fact.**
  - Formal tool use has been sold as “snake oil” in the past.
  - Don’t you need a PhD in mathematics to do this sort of thing?
  - Dubious gains
    - What part of my simulation workload can I not do if I do formal verification?
  - Types of design
    - Much of my logic is data path, and formal tools aren’t good for data path.
    - My design is too big.
  - Assertion languages are seen as adding little or no value by some.
    - Why would I code my design up again in another language? I‘ve got more important things to do.
    - Because not all blocks have assertions, we can’t use formal tools.
  - A long ramp-up time is required to get something running.

REAL (?) TECHNICAL

- **How do I know what I’ve achieved?**

- **How do I review it to make sure it is really doing what it should be doing?**

- **My designs really are too big.**

- **Simulation-based verification people are the norm in the industry, but they don’t have the knowledge/skills to go straight to using formal tools.**
  - Many people don’t know how to extract value from these formal tools.

- **It is very dependent on how good your assertions are, but there is no automated way to evaluate this.**

REAL (?) TECHNICAL

- **It is detached from how we do RTL/software co-verification.**
  - Our simulation is usually driven from real software.

- **Haven’t been able to prove significant things.**
  - Not good at getting feedback on things that it can’t prove.

- **Unproven/unreachable.**
  - There are unproven/reachable cases where I think that it has got past all the states, but it can’t prove anything, and there seems to be nothing that can help you understand why.
  - Is this an issue with the tool, or do we not really understand things?

THE BOTTOM LINE …

- **It isn’t a high enough priority!!!**

- **Brave decision to tell my boss**
  - That I will stop doing some UVM/simulation work to focus on formal verification.
  - That I need a new member of staff to do formal verification.

- **Metrics**
  - Management tends to focus on and prioritize metrics.
    - You can measure code/functional coverage for simulation.
    - You can measure timing.
    - You can measure area.
    - You can measure power.
  - Can you measure formal verification?

QUESTIONS?

Thank You!